

PROGRAM - serverC.c

```
#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<string.h>
#include <stdlib.h>
#include <unistd.h>
void main()
{
    int socketid,b,newsocket,len;
    char recline[100],sendline[100];
    struct sockaddr_in serveraddr,clientaddr;
    bzero(&serveraddr,sizeof(serveraddr));
    //creating socket
    socketid=socket(AF_INET,SOCK_STREAM,0);
    if(socketid<0)
        printf("Socket Cannot be created\n");
    else
        printf("Socket Created\n");
    //setting parameters of socket
    serveraddr.sin_family=AF_INET;
    serveraddr.sin_port=htons(3000);
    serveraddr.sin_addr.s_addr=htonl(INADDR_ANY);
    //binding socket
    bind(socketid,(struct sockaddr*)&serveraddr,sizeof(serveraddr));
    if(b<0)
        printf("Cannot bind socket\n");
    else
        printf("Socket Binded Successfully\n");
    //listen()
    listen(socketid,10);

    len=sizeof(clientaddr);
    newsocket=accept(socketid,(struct sockaddr*)&clientaddr,&len);
    printf("Connection Accepted\n");

    bzero(recline,100);
    bzero(sendline,100);

    while(1)
    {
        //message recv from client
        read(newsocket,recline,100);
        if (strcmp(recline,"quit")==0 || strcmp(sendline,"quit")== 0)
            exit(0);
        //display
        printf("Message from Client = ");
        puts(recline);
        // send msg to client by server
```

```

        printf("Message to Client = ");
        gets(sendline);
        write(newsocket, sendline, sizeof(sendline));
        if (strcmp(recline, "quit")==0 || strcmp(sendline, "quit")== 0)
            exit(0);
    }
    close(socketid);
}

```

PROGRAM – clientC.c

```

#include<stdio.h>
#include<sys/socket.h>
#include<netinet/in.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>
void main()
{
    int socketid, len;
    char sendline[100], recline[100];
    struct sockaddr_in serveraddr, clientaddr;
    bzero(&serveraddr, sizeof(serveraddr));
    //creating socket
    socketid=socket(AF_INET, SOCK_STREAM, 0);
    if(socketid<0)
        printf("Socket Cannot be created\n");
    else
        printf("Socket Created\n");
    //setting parameters of socket
    serveraddr.sin_family=AF_INET;
    serveraddr.sin_port=htons(3000);
    serveraddr.sin_addr.s_addr=htonl(INADDR_ANY);
    //connecting to server
    connect(socketid, (struct sockaddr*)&serveraddr, sizeof(serveraddr));
    printf("Client Connected\n");

    bzero(sendline, 100);
    bzero(recline, 100);

    while(1)
    {
        if(strcmp(recline, "quit")==0 || strcmp(sendline, "quit")==0)
            exit(0);
        //messaging from client to server
        printf("Message to Server = ");
        gets(sendline);
        write(socketid, sendline, sizeof(sendline));
        //message recv and display from server
        read(socketid, recline, 100);
        if(strcmp(recline, "quit")==0 || strcmp(sendline, "quit")==0)

```

```

        exit(0);
    printf("Message from Server = ");
    puts(recline);
}
    close(socketid);
}

```

OUTPUT

```

admin_GK@administrator52: ~/Desktop
admin_GK@administrator52:~/Desktop$ gcc clientC.c
clientC.c: In function 'main':
clientC.c:40:17: warning: implicit declaration of function 'gets'; did you mean
'fgets'? [-Wimplicit-function-declaration]
   40 |             gets(sendline);
      |             ^~~~~
      |             fgets
/usr/bin/ld: /tmp/cczeSfqg.o: in function `main':
clientC.c:(.text+0x174): warning: the `gets' function is dangerous and should not
be used.
admin_GK@administrator52:~/Desktop$ ./a.out
Socket Created
Client Connected
Message to Server = hi
Message from Server = hello
Message to Server = bye
Message from Server = ok bye
Message to Server = quit
admin_GK@administrator52:~/Desktop$

```

```

admin_GK@administrator52: ~/Desktop
admin_GK@administrator52:~/Desktop$ gcc serverC.c
serverC.c: In function 'main':
serverC.c:57:17: warning: implicit declaration of function 'gets'; did you mean
'fgets'? [-Wimplicit-function-declaration]
   57 |             gets(sendline);
      |             ^~~~~
      |             fgets
/usr/bin/ld: /tmp/ccSCvZoR.o: in function `main':
serverC.c:(.text+0x219): warning: the `gets' function is dangerous and should not
be used.
admin_GK@administrator52:~/Desktop$ ./a.out
Socket Created
Socket Binded Successfully
Connection Accepted
Message from Client = hi
Message to Client = hello
Message from Client = bye
Message to Client = ok bye
admin_GK@administrator52:~/Desktop$

```